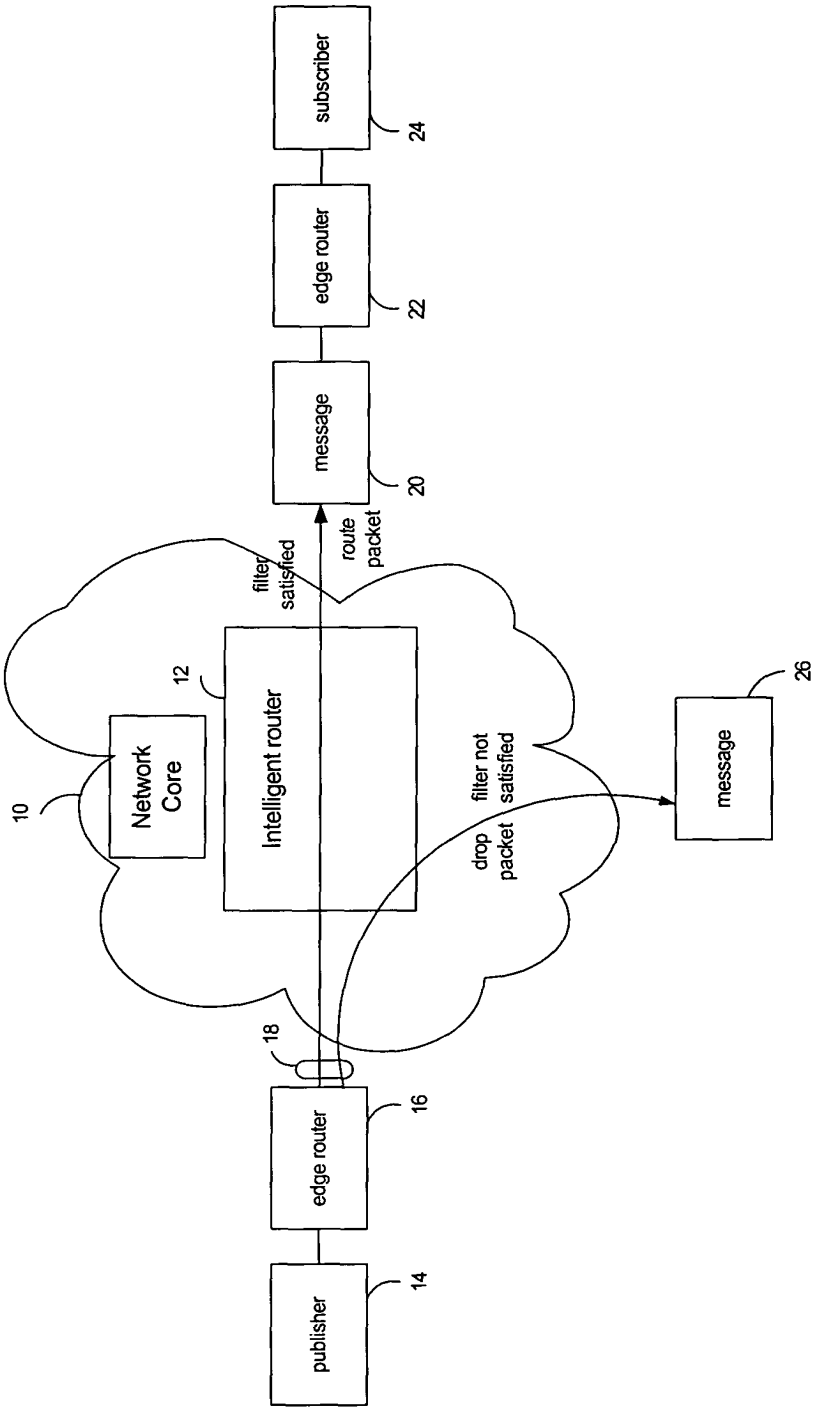
subscriber
24
edge router
22
message
20
route packet
filter satisfied
12
Intelligent router
Network Core
10
filter not satisfied
drop packet
message
26
18
edge router
16
publisher
14

FIG. 1

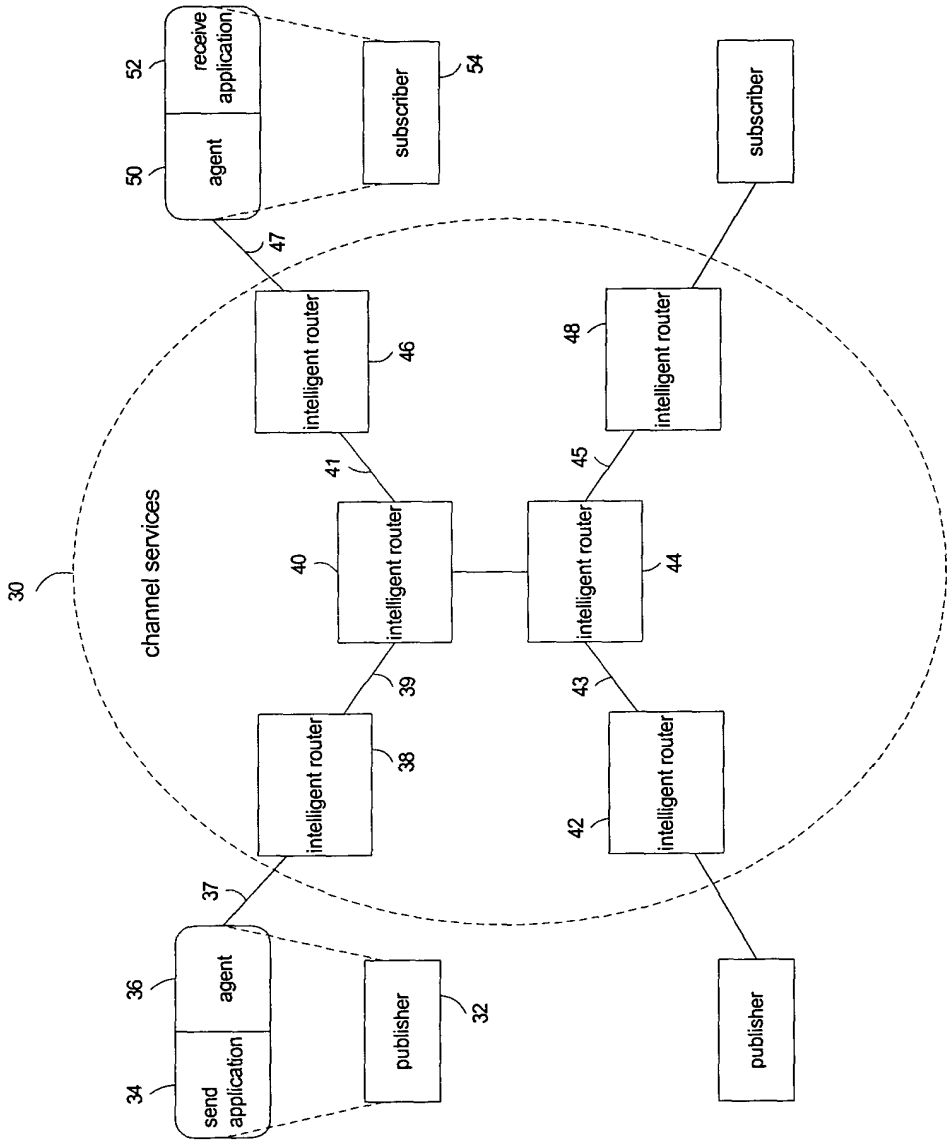
30
channel services
52
receive application
50
agent
47
subscriber
54
subscriber
intelligent router
46
intelligent router
48
41
45
intelligent router
40
intelligent router
44
39
43
intelligent router
38
intelligent router
42
37
36
agent
34
send application
publisher
32
publisher

FIG. 2

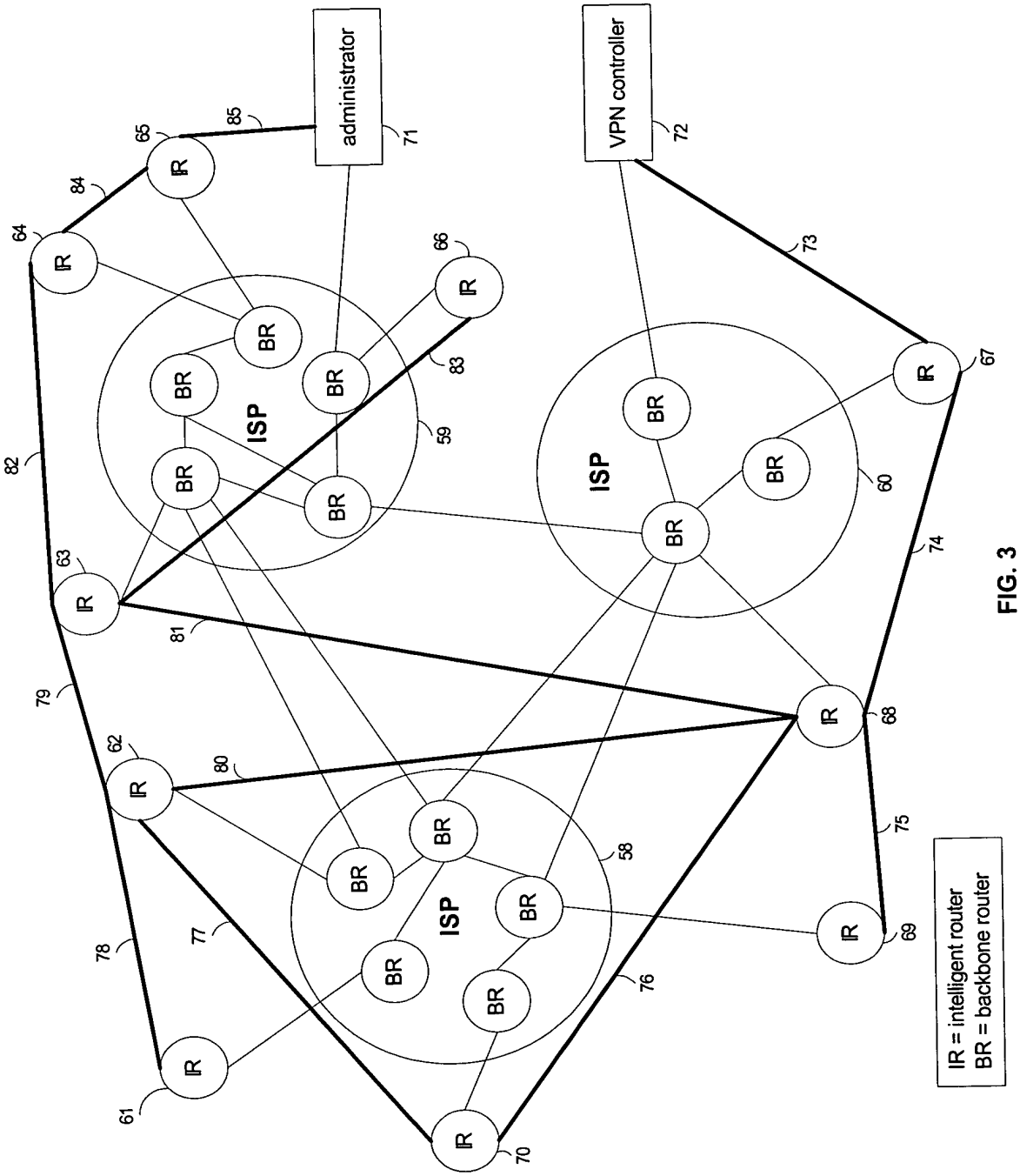
administrator
71
VPN controller
72
ISP
59
ISP
60
ISP
58
IR
61
IR
62
IR
63
IR
64
IR
65
IR
66
IR
67
IR
68
IR
69
IR
70
BR
73
74
75
76
77
78
79
80
81
82
83
84
85
IR = intelligent router
BR = backbone router

FIG. 3

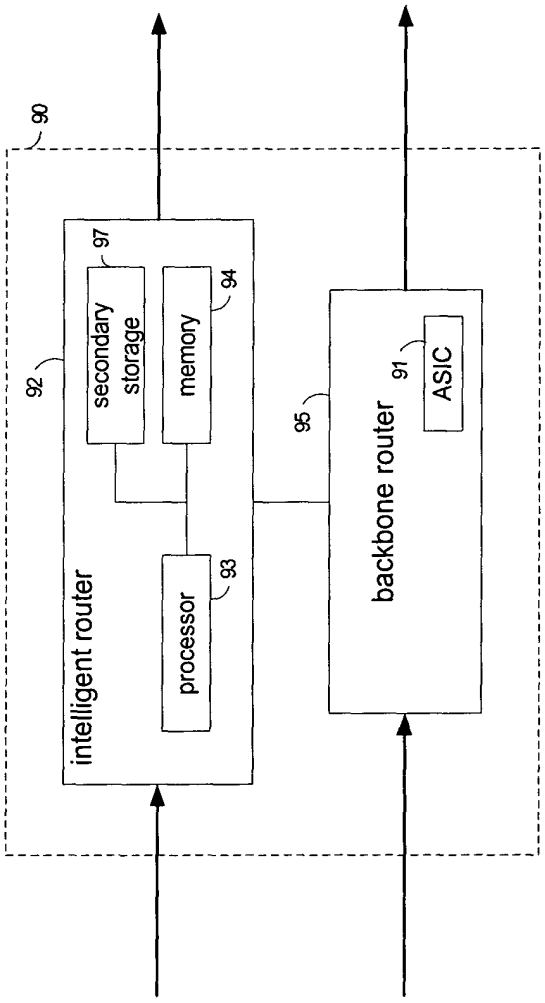
90
97
secondary storage
94
memory
92
95
91
ASIC
backbone router
intelligent router
93
processor

FIG. 4

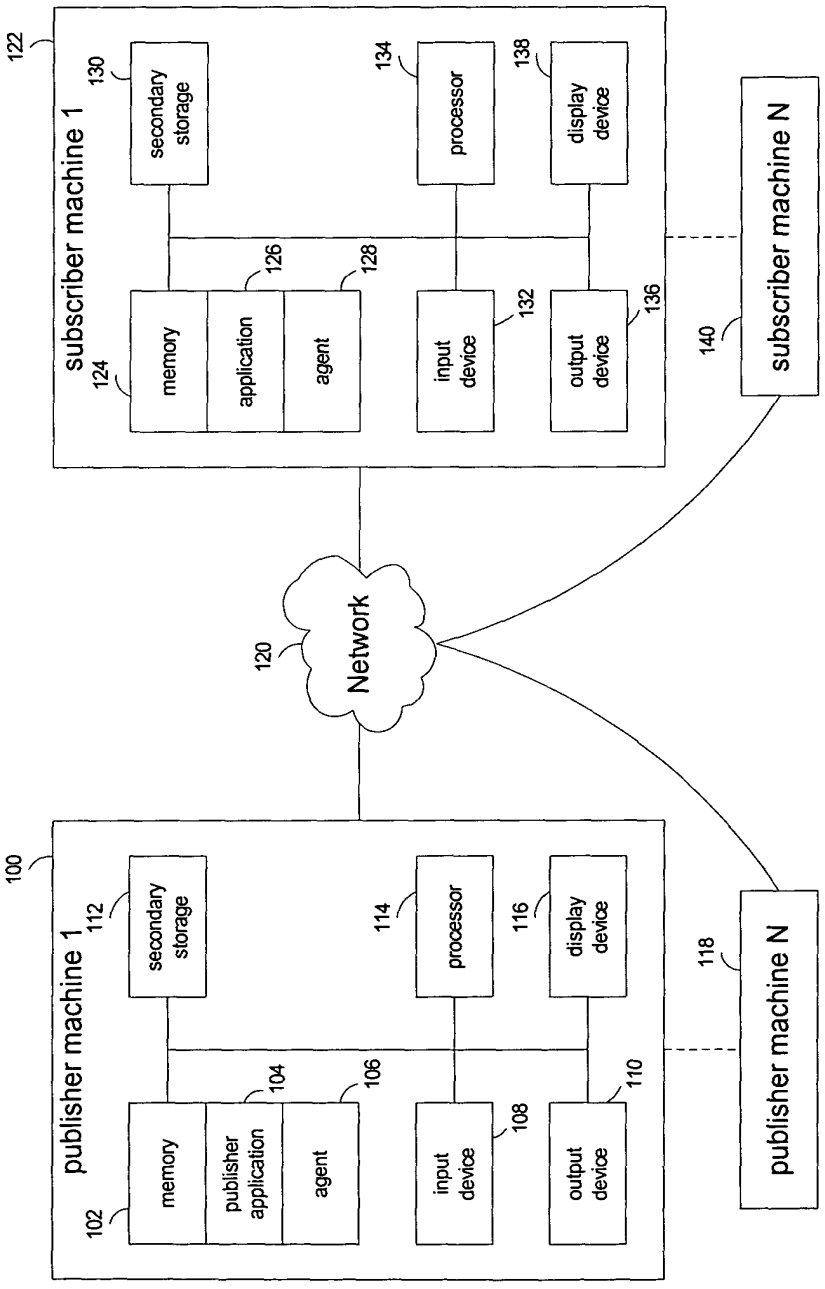
100
publisher machine 1
102
memory
104
publisher application
106
agent
108
input device
110
output device
112
secondary storage
114
processor
116
display device
118
publisher machine N
120
Network
122
subscriber machine 1
124
memory
126
application
128
agent
130
secondary storage
132
input device
134
processor
136
output device
138
display device
140
subscriber machine N

FIG. 5

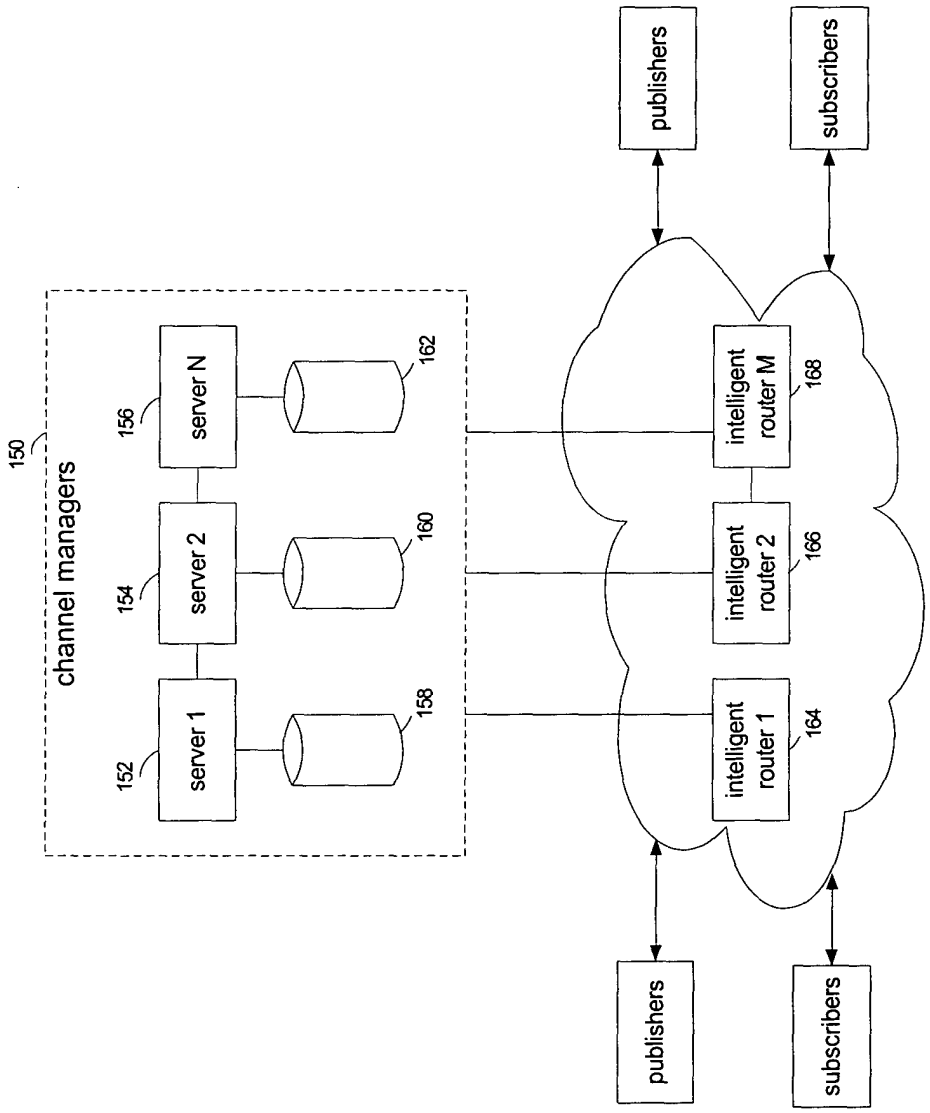
150
channel managers
server 1
152
server 2
154
server N
156
158
160
162
intelligent router 1
164
intelligent router 2
166
intelligent router M
168
publishers
subscribers
publishers
subscribers

FIG. 6

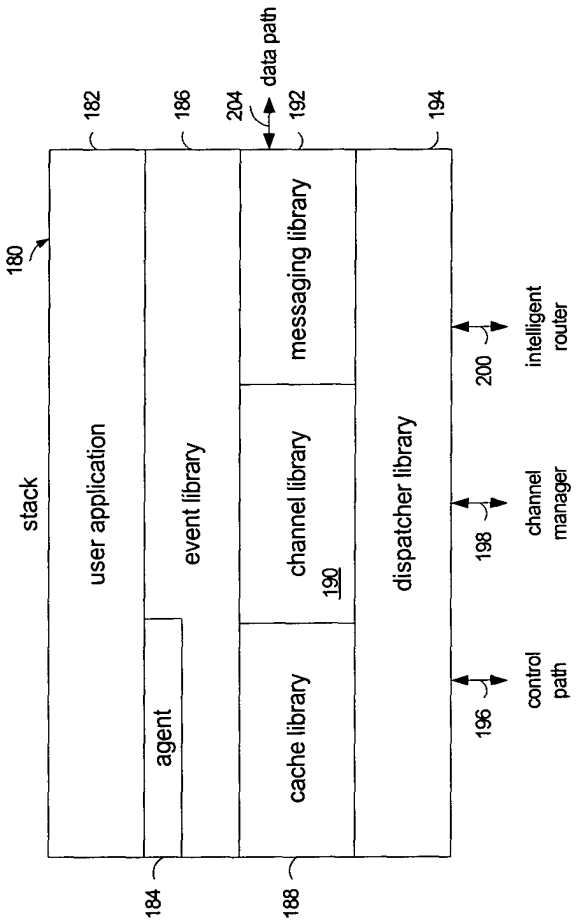
180
stack
user application
182
agent
184
event library
186
cache library
188
channel library
190
messaging library
192
data path
204
dispatcher library
194
control path
196
channel manager
198
intelligent router
200

FIG. 7

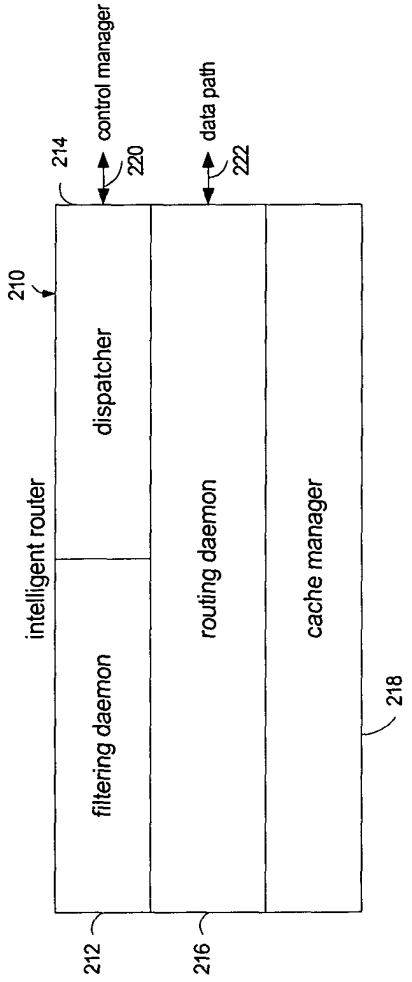
control manager
data path
214
220
222
210
dispatcher
intelligent router
routing daemon
cache manager
filtering daemon
218
212
216

FIG. 8

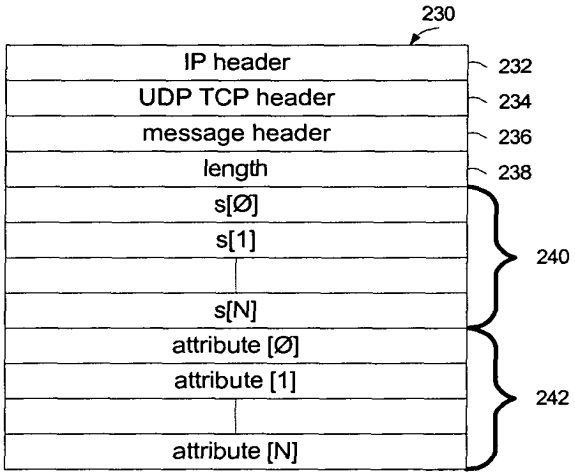
230
IP header
232
UDP TCP header
234
message header
236
length
238
s[Ø]
s[1]
240
s[N]
attribute [Ø]
attribute [1]
242
attribute [N]

FIG. 9

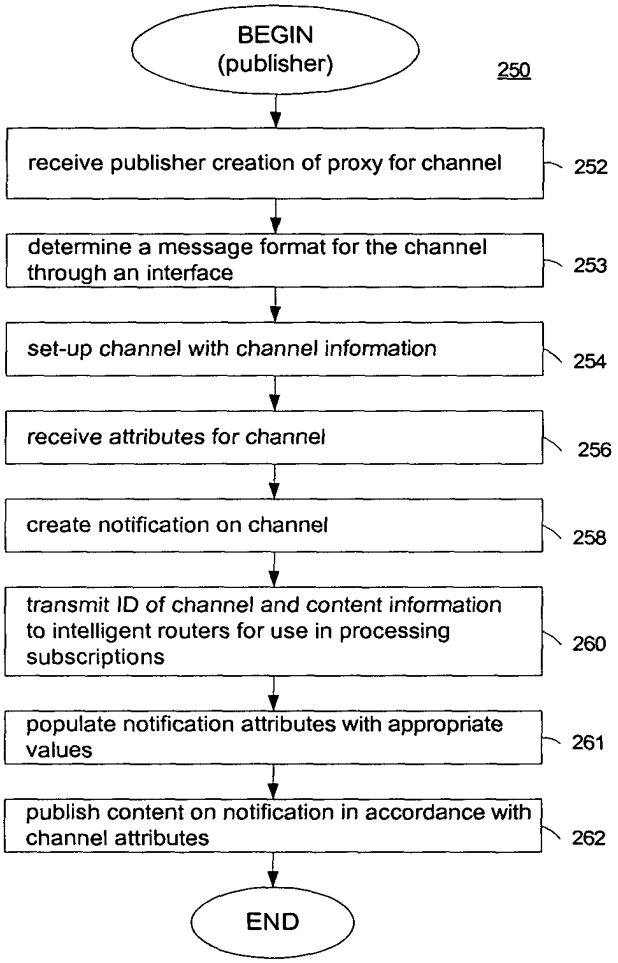
BEGIN
(publisher)
250
receive publisher creation of proxy for channel
252
determine a message format for the channel through an interface
253
set-up channel with channel information
254
receive attributes for channel
256
create notification on channel
258
transmit ID of channel and content information to intelligent routers for use in processing subscriptions
260
populate notification attributes with appropriate values
261
publish content on notification in accordance with channel attributes
262
END

FIG. 10

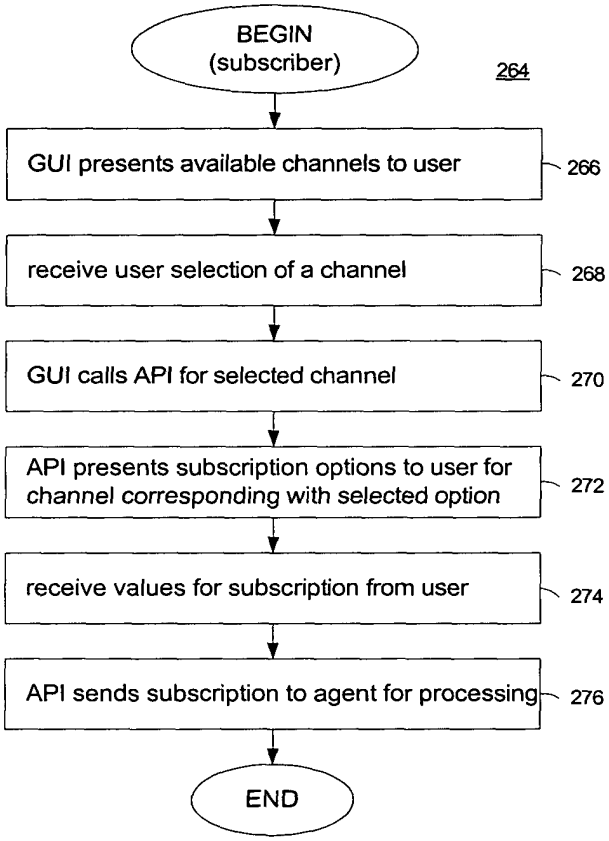
BEGIN
(subscriber)
264
GUI presents available channels to user
266
receive user selection of a channel
268
GUI calls API for selected channel
270
API presents subscription options to user for channel corresponding with selected option
272
receive values for subscription from user
274
API sends subscription to agent for processing
276
END

FIG. 11

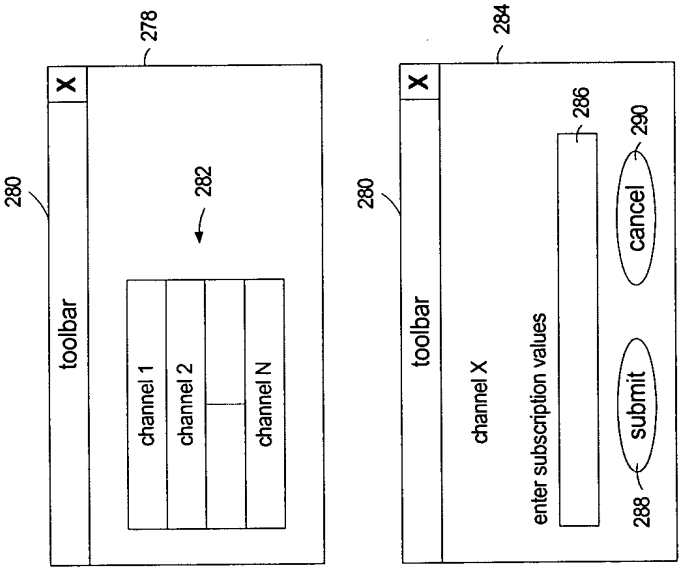
278
280
X
toolbar
282
channel 1
channel 2
channel N
284
280
X
toolbar
channel X
enter subscription values
286
290
cancel
submit
288

FIG. 12

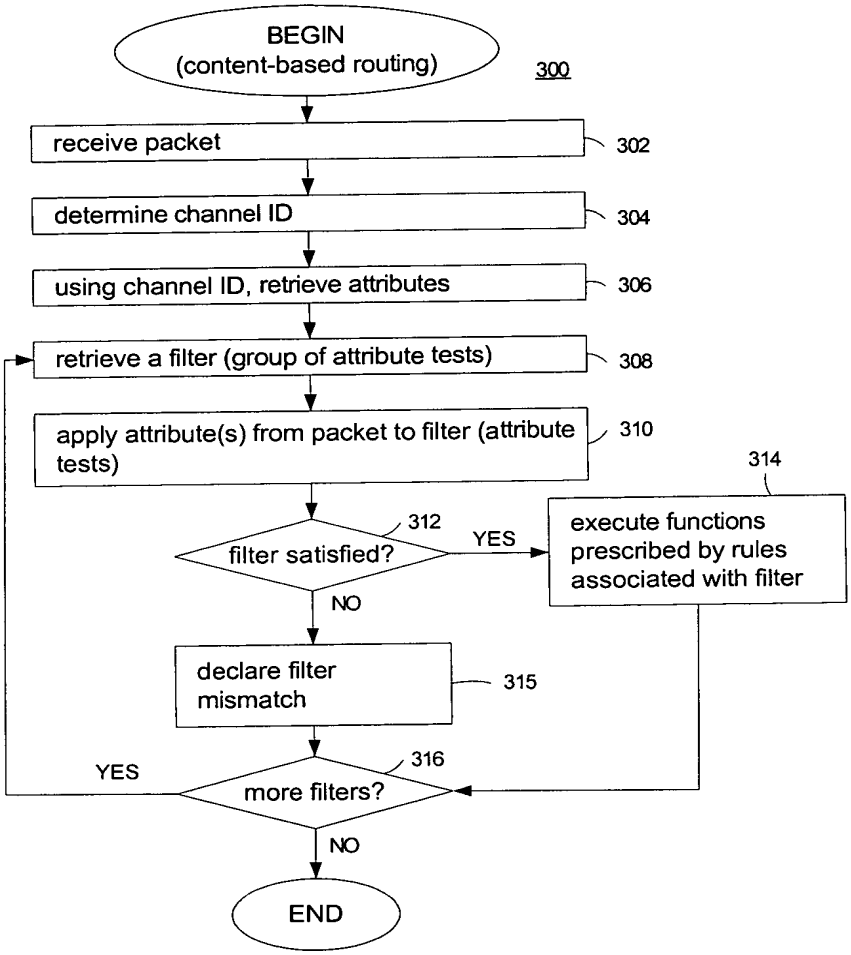
BEGIN
(content-based routing)
300
receive packet
302
determine channel ID
304
using channel ID, retrieve attributes
306
retrieve a filter (group of attribute tests)
308
apply attribute(s) from packet to filter (attribute tests)
310
314
execute functions prescribed by rules associated with filter
312
filter satisfied?
YES
NO
declare filter mismatch
315
316
more filters?
YES
NO
END

FIG. 13

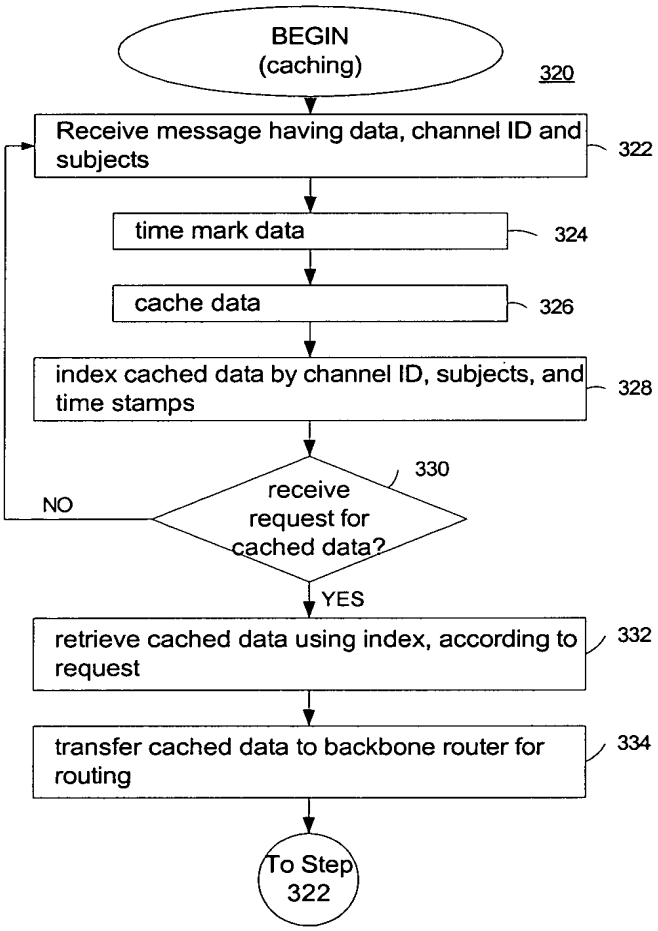
BEGIN
(caching)
320
Receive message having data, channel ID and subjects
322
time mark data
324
cache data
326
index cached data by channel ID, subjects, and time stamps
328
330
receive request for cached data?
NO
YES
retrieve cached data using index, according to request
332
transfer cached data to backbone router for routing
334
To Step 322

FIG. 14

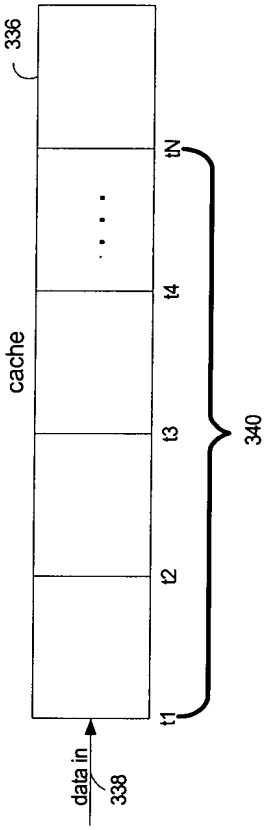

FIG. 15

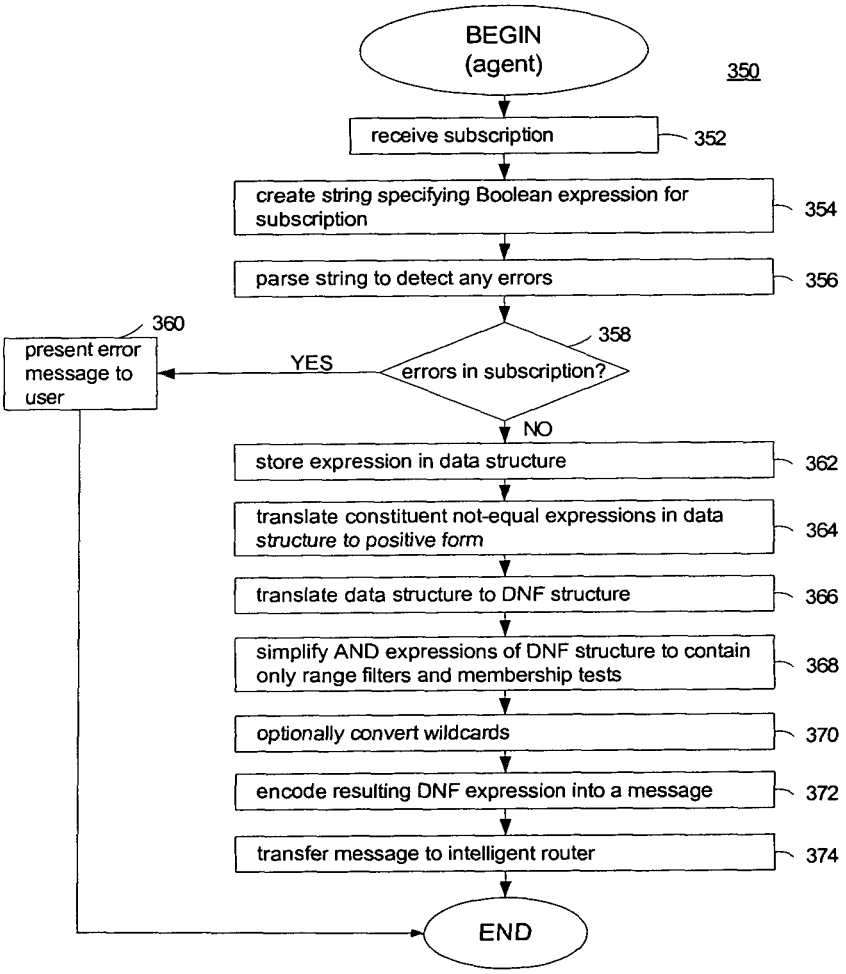
BEGIN
(agent)
350
receive subscription
352
create string specifying Boolean expression for subscription
354
parse string to detect any errors
356
358
errors in subscription?
YES
360
present error message to user
NO
store expression in data structure
362
translate constituent not-equal expressions in data structure to positive form
364
translate data structure to DNF structure
366
simplify AND expressions of DNF structure to contain only range filters and membership tests
368
optionally convert wildcards
370
encode resulting DNF expression into a message
372
transfer message to intelligent router
374
END

FIG. 16

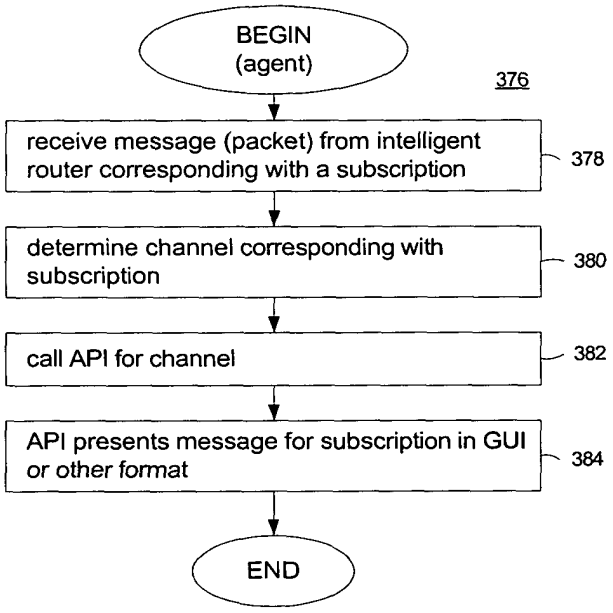
BEGIN
(agent)
376
receive message (packet) from intelligent router corresponding with a subscription
378
determine channel corresponding with subscription
380
call API for channel
382
API presents message for subscription in GUI or other format
384
END

FIG. 17

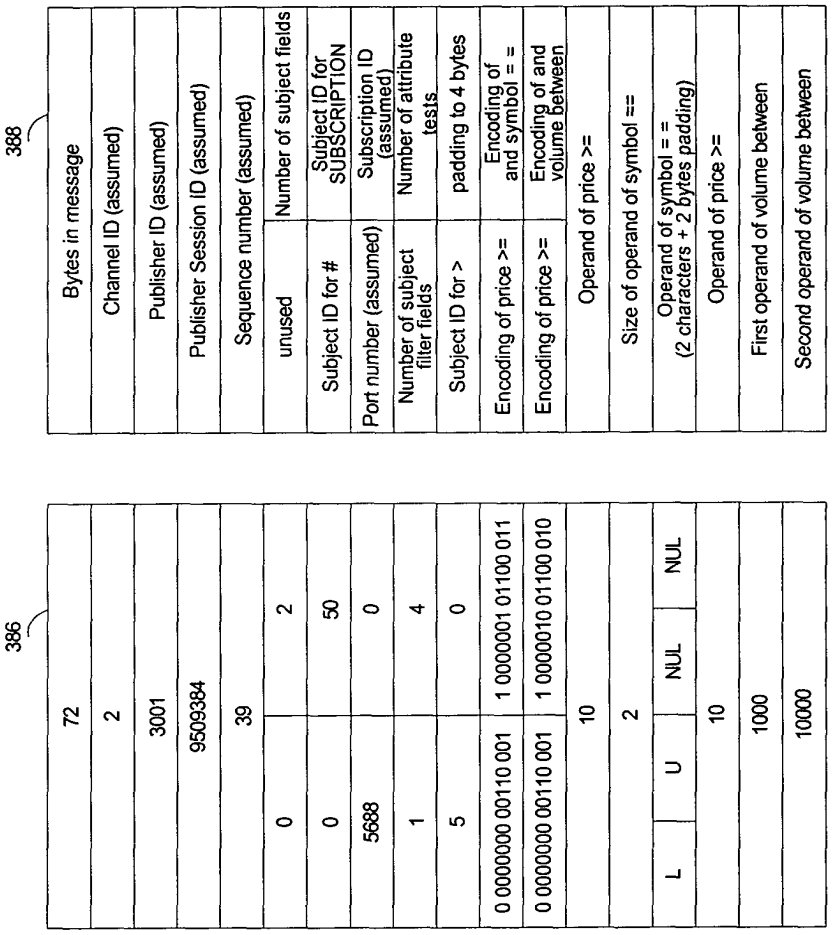
386
72
2
3001
9509384
39
0
2
0
50
5688
0
1
4
5
0
0 0000000 00110 001
1 0000001 01100 011
0 0000000 00110 001
1 0000010 01100 010
10
2
L
U
NUL
NUL
10
1000
10000
388
Bytes in message
Channel ID (assumed)
Publisher ID (assumed)
Publisher Session ID (assumed)
Sequence number (assumed)
unused
Number of subject fields
Subject ID for #
Subject ID for SUBSCRIPTION
Port number (assumed)
Subscription ID (assumed)
Number of subject filter fields
Number of attribute tests
Subject ID for >
padding to 4 bytes
Encoding of price >=
Encoding of and symbol = =
Encoding of price >=
Encoding of and volume between
Operand of price >=
Size of operand of symbol ==
Operand of symbol = = (2 characters + 2 bytes padding)
Operand of price >=
First operand of volume between
Second operand of volume between

FIG. 18

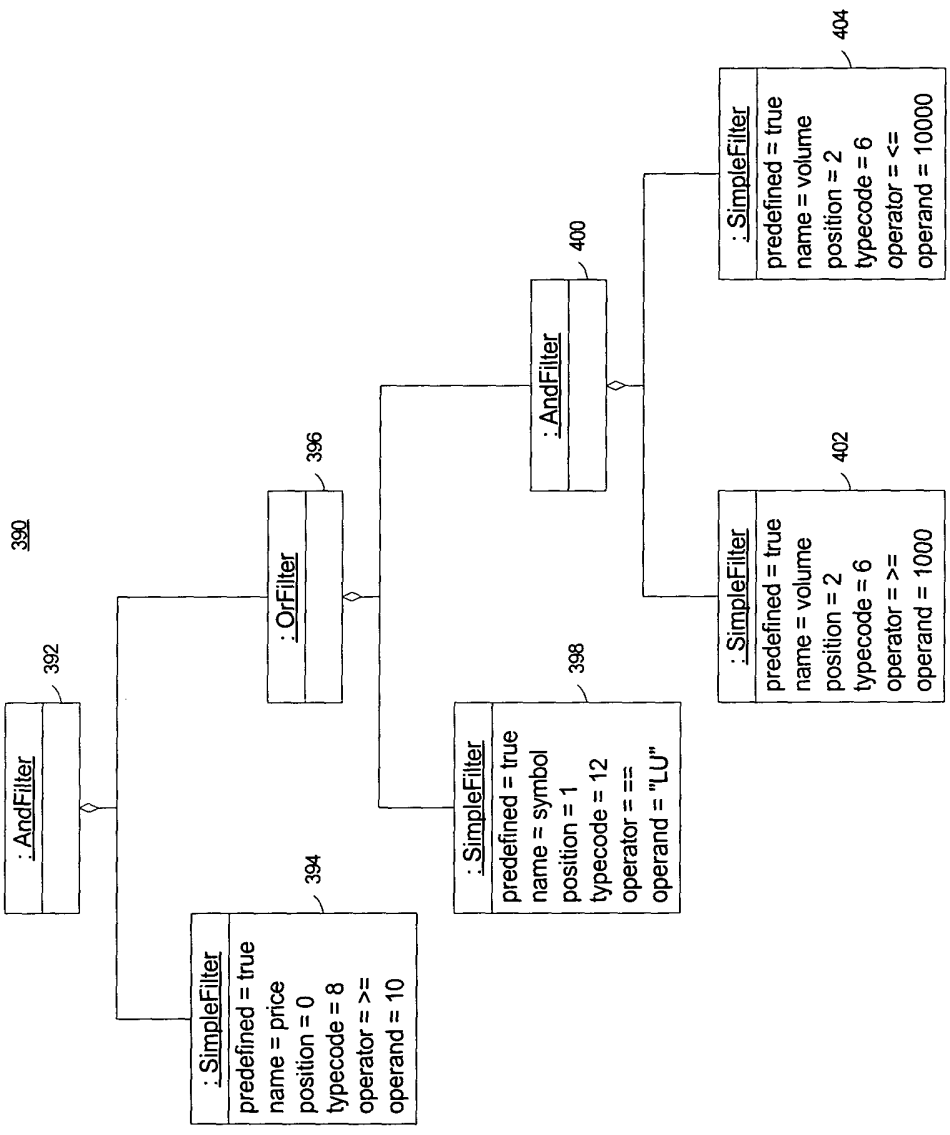
390
: AndFilter
392
: SimpleFilter
predefined = true
name = price
position = 0
typecode = 8
operator = >=
operand = 10
394
: OrFilter
396
: SimpleFilter
predefined = true
name = symbol
position = 1
typecode = 12
operator = ==
operand = "LU"
398
: AndFilter
400
: SimpleFilter
predefined = true
name = volume
position = 2
typecode = 6
operator = >=
operand = 1000
402
: SimpleFilter
predefined = true
name = volume
position = 2
typecode = 6
operator = <=
operand = 10000
404

FIG. 19

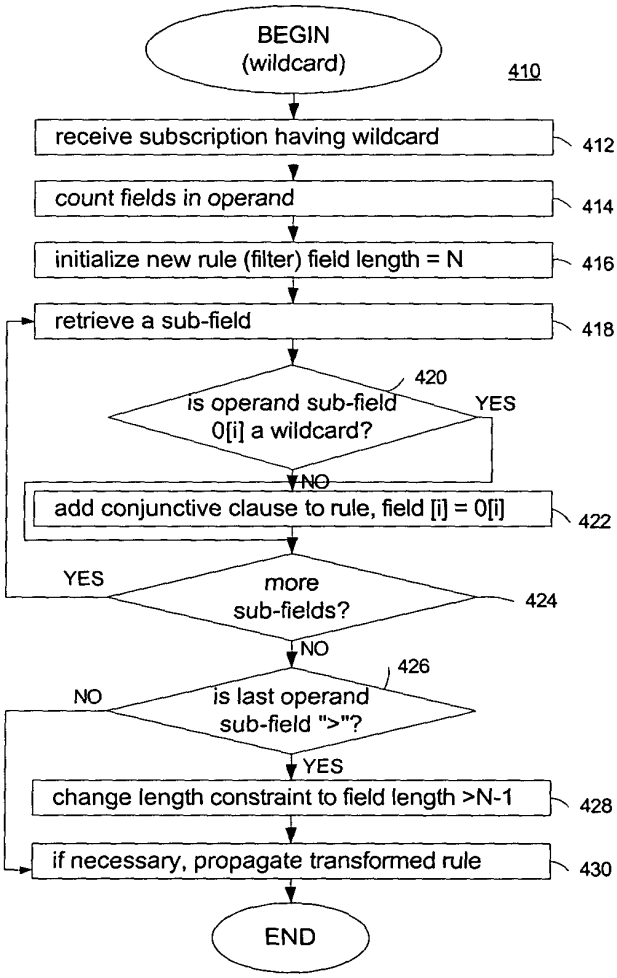
BEGIN
(wildcard)
410
receive subscription having wildcard
412
count fields in operand
414
initialize new rule (filter) field length = N
416
retrieve a sub-field
418
420
is operand sub-field 0[i] a wildcard?
YES
NO
add conjunctive clause to rule, field [i] = 0[i]
422
YES
more sub-fields?
424
NO
426
NO
is last operand sub-field ">"?
YES
change length constraint to field length >N-1
428
if necessary, propagate transformed rule
430
END

FIG. 20

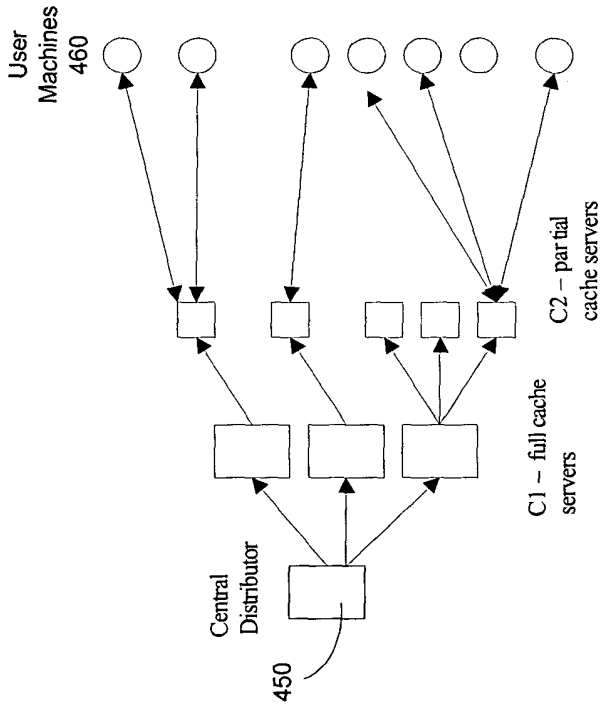
User Machines 460
C2 – partial cache servers
C1 – full cache servers
Central Distributor
450

FIG. 21

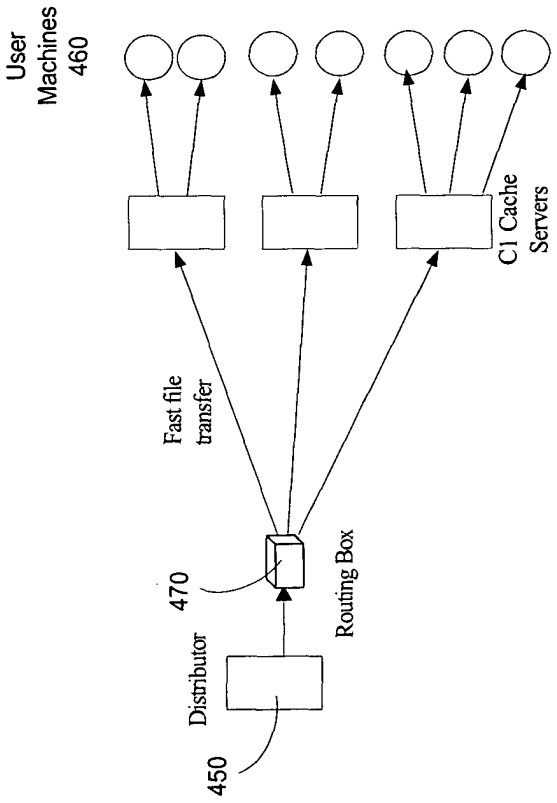
User Machines 460
C1 Cache Servers
Fast file transfer
470
Routing Box
Distributor
450

FIG. 22

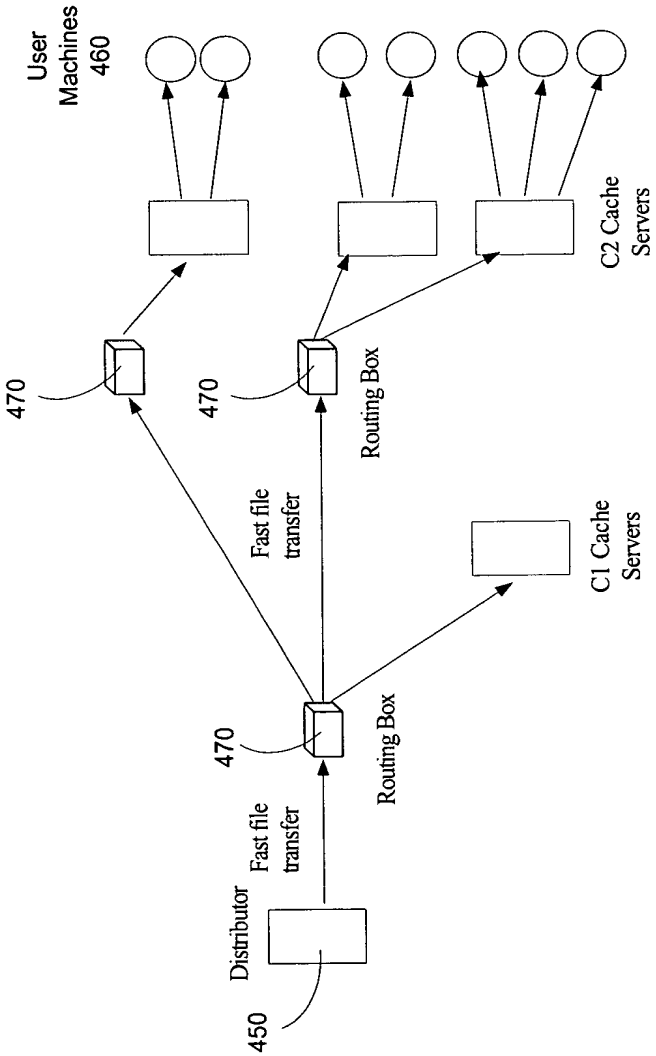
User Machines 460
C2 Cache Servers
C1 Cache Servers
Routing Box
Routing Box
Routing Box
Fast file transfer
Fast file transfer
Distributor
470
470
470
450

FIG. 23